

LATVIJAS UNIVERSITĀTE
DATORIKAS FAKULTĀTE

**KAMERAS KONTROLES PROGRAMMATŪRA UN CHROME SPRAUDNIS
SISTĒMAS STĀVOKĻA PĀRRAUDZĪBAI UN KONTROLEI**

KVALIFIKĀCIJAS DARBS

Autors: **Kārlis Punka**

Studenta apliecības Nr.: kp13050

Darba vadītājs: B. dat. Nils Lindentāls

RĪGA 2015

ANOTĀCIJA

Kvalifikācijas darba mērķis ir izstrādāt programmatūru fotoaparāta *Sony QX1*, vai citu *Sony Remote Camera API* izmantojošu kameru attālinātai kontrolei bezvadu tīklā, grafisko vidi nodrošinot ar tīmekļa pārlūka *Google Chrome* spraudni. Šāds risinājums ļautu nodrošināt programmatūras darbību neatkarīgi no lietotāja izvēlētās personālā datora operētājsistēmas.

Sistēma izstrādāta, izmantojot tīmekļa tehnoloģijas un programmēšanas valodu Python. Sistēmas lietotāji var attālināti pārraudzīt kameras uzņemšanas parametrus un veikt attēla uzņemšanu.

Atslēgas vārdi: pārlūkprogrammas spraudnis, Python, Sony Remote Camera API

ABSTRACT

The aim of this qualification work described in this document is to develop a set of software for remote control of cameras using *Sony Remote Camera API*, such as *Sony QX1* over wireless network, without constraining the user to use a specific operating system.

This is achieved by providing the graphical interface in the browser by developing an extension for *Google Chrome* and a supporting application that is developed using *Python* programming language.

Keywords: Browser extension, Python, Sony Remote Camera API

Satura rādītājs

| | |
|--|-----------|
| 1. Programmatūras prasību specifikācija | 5 |
| 1.1. Ievads..... | 5 |
| 1.1.1 Prasību specifikācijas nolūks | 5 |
| 1.1.2. Darbības sfēra | 5 |
| 1.1.3. Saistība ar citiem dokumentiem..... | 5 |
| 1.1.4. Definīcijas, akronīmi un saīsinājumi | 5 |
| 1.1.5. Dokumenta pārskats..... | 5 |
| 1.2. Vispārējs apraksts | 6 |
| 1.2.1. Produkta perspektīva..... | 6 |
| 1.2.2. Produkta funkcijas | 6 |
| 1.2.3 Sistēmas lietotāji un to raksturojumi..... | 6 |
| 1.3. Funkcionālās prasības | 6 |
| 1.3.1. Nodrošināt savienojumu ar kameru | 6 |
| 1.3.3. Attālināti uzņemt attēlus ar kameru | 7 |
| 1.4. Ārējā saskarne..... | 8 |
| 1.4.1. Lietotāja saskarne | 8 |
| 1.4.2. Aparatūras saskarne | 8 |
| 1.4.3. Programmatūras saskarne | 8 |
| 2. Programmatūras projektējuma apraksts..... | 9 |
| 2.1. Ievads..... | 9 |
| 2.1.1. Projektējuma apraksta nolūks | 9 |
| 2.1.2. Darbības sfēra | 9 |
| 2.1.3. Saistība ar citiem dokumentiem..... | 9 |
| 2.2. Dekompozīcijas apraksts | 9 |
| 2.2.1. Grafiskā vide..... | 9 |
| 2.2.2. Spraudnis pārlūkprogrammā..... | 9 |
| 2.2.3. Fotoaparāta kontroles servera uzbūve..... | 10 |
| 2.2.4. Fotoaparāta saskarne..... | 10 |
| 2.2.5. Žurnālēšanas saskarnes modulis | 10 |
| 2.3. Datu struktūras un formāti | 11 |
| 2.3.1. Datu struktūra datu apmaiņai starp pārlūkprogrammatūru un serveri | 11 |
| 2.3.1.3. Kameras stāvokļa atbilde | 13 |
| 2.3.2. Datu struktūra datu apmaiņai starp serveri un kameru | 22 |
| 2.4 Moduļu projektējums | 24 |
| 2.4.1 Pārlūka spraudnis..... | 24 |
| 2.4.2. Servera moduļi..... | 27 |

| | |
|---|-----------|
| 2.5. Lietotāja saskarnes apraksts | 35 |
| 2.5.1 Skats “Sākt darbu” | 35 |
| 2.5.2 Skats “Darbības ar kameru” | 36 |
| 3. Testēšanas dokumentācija | 38 |
| 3.1. ChromeCommunication moduļa testi | 38 |
| 3.1.1. send_message funkcijas testi | 38 |
| 3.1.2. read_input funkcijas testi | 38 |
| 3.1.3. process_message funkcijas testi | 39 |
| 3.1.4. process_message_all funkcijas testi | 40 |
| 3.2. cameraCommunication moduļa testi | 40 |
| 3.2.1. get_camera_xml_address funkcijas testi | 40 |
| 3.2.2. fetch_camera_access_urls funkcijas testi | 41 |
| 3.2.4. is_camera_connected funkcijas testi | 41 |
| 3.2.5. convert_parameter_request_boolean_values funkcijas testi | 42 |
| evaluate_request_success funkcijas testi | 42 |
| 3.2.7. send_request_to_camera funkcijas testi | 43 |
| 3.2.8. connect_to_camera funkcijas testi | 43 |
| 3.2.9. get_current_state funkcijas testi | 44 |
| 3.3. cameraResponseParser moduļa testi | 44 |
| 3.3.1. parse_current_state_response funkcijas testi | 44 |
| 3.3.2. parse_connection_status_request funkcijas testi | 45 |
| 3.3.3. parse_init_connection_request funkcijas testi | 45 |
| 3.3.4. act_take_picture funkcijas testi | 46 |
| 4. Projekta organizācija | 47 |
| 5. Konfigurācijas pārvaldība | 48 |
| 6. Kvalitātes nodrošināšana | 49 |
| 7. Darbietilpības novērtējums | 50 |
| 7.1. COCOMO aprēķina metode | 50 |
| 7.2. Faktiskā laika patēriņš | 51 |
| 8. Nobeigums | 55 |
| 9. Definīcijas, akronīmi un saīsinājumi | 56 |
| Pirmkods | 57 |
| Izmantotā literatūra | 59 |

1. Programmatūras prasību specifikācija

1.1. Ievads

Šī dokumenta nodaļa (turpmāk PPS) ir dokuments, kas balstīts uz LVS 68:1996 ceļvedi programmatūras prasību specifikāciju rakstīšanai.

1.1.1 Prasību specifikācijas nolūks

Šīs prasību specifikācijas nolūks ir precīzi, viennozīmīgi un kodolīgi aprakstīt izstrādājamās kameras kontroles sistēmas prasības. Šī specifikācija ir paredzēta, lai pēc iespējas labāk nodrošinātu savstarpējo sapratni starp pasūtītāju un izstrādātāju un nerastos pārpratumi par sistēmas izstrādājamajām funkcijām.

Šis projekts tiks izstrādāts un tālākā dokumentācija tiks izstrādāta, vadoties pēc šīs specifikācijas. Sistēmas izstrādē tiks realizētas šajā specifikācijā norādītās funkcijas.

Prasību specifikācija ir nepieciešama pasūtītājam un izstrādātājam.

1.1.2. Darbības sfēra

Kameras kontroles sistēma ir paredzēta galalietotāja personīgai lietošanai apvienojumā ar *Sony* piedāvāto kameru *QXI*. Šīs sistēmas lietotāji var būt gan foto entuziasti, gan foto nozares profesionāļi, kam nepieciešams veikt foto uzņemšanu attālināti, neesot tiešā sasniedzamībā fotoaparātam.

1.1.3. Saistība ar citiem dokumentiem

Specifikācija ir balstīta uz “Latvijas standarts” standartu 68:1996 “Programmatūras specifikāciju ceļvedis”.

1.1.4. Definīcijas, akronīmi un saīsinājumi

Definīcijas, akronīmus un saīsinājumus var aplūkot darba beigās, sadaļā “Definīcijas, akronīmi un saīsinājumi”.

1.1.5. Dokumenta pārskats

Dokuments tiek veidots, daļēji vadoties pēc LVS 68:1996 standarta “Programmatūras specifikācijas ceļvedis” norādītajām vadlīnijām. Specifikācijas struktūru veido piecas sadaļas:

Ievads – raksturo dokumenta struktūru, mērķi, iemeslu, kādēļ tas ir sastādīts. Ievadā tiek īsi aprakstīta projekta būtība, kādēļ sistēma varētu tikt veidota un kāda funkcionalitāte var tikt vai netikt iekļauta.

Vispārējs apraksts – vispārīgā formā apraksta projektu. Aplūko projekta funkcionalitāti, klienta īpatnības, kā arī sistēmas ierobežojumus.

Funkcionālās prasības – raksturo vispārīgajā aprakstā minētās funkcionalitāti (ievaddati, sagaidāmās darbības, izvaddatus). Šī sadaļa satur funkciju specifikāciju.

Ārējā saskarne – raksturo projekta prasības sistēmas lietotāja saskarnei, aparatūras saskarnei un programmatūras saskarnei.

1.2. Vispārējs apraksts

1.2.1. Produkta perspektīva

Sistēma būs tīmekļa tehnoloģijās balstīta sistēma, kas atradīsies un tiks darbināta no galalietotāja datora. Sistēmas lietošanai datoram būs nepieciešams bezvadu tīkla adapteris un iespēja piekļūt kameras piedāvātajam bezvadu tīklam. Sistēmas galalietotājs var būt jebkura ieinteresēta persona.

1.2.2. Produkta funkcijas

1. Nodrošināt savienojumu ar kameru
2. Iegūt un attēlot kameras aktuālos darbības parametrus.
3. Attālināti uzņemt attēlus ar kameru.

1.2.3 Sistēmas lietotāji un to raksturiezīmes

Sistēmai būs 2 lietotāju grupas:

- Lietotāji bez priekšzināšanām foto uzņemšanā, kas ir ieinteresēti veikt attālinātu foto uzņemšanu bez uzņemšanas parametru kontroles. Šiem lietotājiem ir nepieciešamas pamatzināšanas par datora un pārlūkprogrammas lietošanu.
- Lietotāji ar priekšzināšanām foto uzņemšanā, kas būs ieinteresēti attālinātā foto uzņemšanā ar uzņemšanas parametru kontroli. Šiem lietotājiem ir nepieciešamas pamatzināšanas par datora un pārlūkprogrammas lietošanu, kā arī zināšanām par foto uzņemšanu.

1.3. Funkcionālās prasības

1.3.1. Nodrošināt savienojumu ar kameru

Mērķis:

Izveidot sākotnējo savienojumu starp programmatūru un kameru, lai būtu iespējams veikt tālākās darbības.

Ievaddati: Redzot skatu “Pieslēgties kamerai”, lietotājs izvēlas izveidot savienojumu ar kameru, un aktivizē pogu “Pieslēgties”.

Darbības:

Sistēma uzsāk savienojuma izveidošanu ar kameru, un gaida datus ar informāciju par pieslēgumu kamerai.

Izvaddati:

Sistēma, sagaidot datus ar stāvokli “Ir pieslēgums”, aktīvajā skatā attēlo paziņojumu par sekmīgu savienojuma nodibināšanu un nomaina skatu uz “Darbības ar kameru”. Sistēma,

sagaidot datus ar stāvokli “Neizdevās nodibināt pieslēgumu”, aktīvajā skatā attēlo kļūdas paziņojumu par nesekmīgu savienojuma nodibināšanu.

1.3.2. Nodrošināt iespēju iegūt kameras aktuālos darbības parametrus.

Mērķis:

Iegūt kameras stāvokļa atskaiti un attēlot to aktīvajā skatā.

Ievaddati: Redzot skatu “Darbības ar kameru”, lietotājs izvēlas iegūt kameras stāvokļa atskaiti, un aktivizē pogu “Atskaite par kameru”.

Darbības:

Sistēma nosūta pieprasījumu kamerai un gaida datus ar informāciju ar kameras stāvokļa atskaiti.

Izvaddati:

Sistēma, sagaidot datus ar kameras stāvokļa atskaiti, aktīvajā skatā attēlo paziņojumu par sekmīgu atskaites iegūšanu un attēlo atskaiti aktīvajā skatā. Sistēma, sagaidot datus ar stāvokli “Nav savienojums ar kameru”, aktīvajā skatā attēlo kļūdas paziņojumu.

1.3.3. Attālināti uzņemt attēlus ar kameru.

Mērķis:

Veikt kamerai pieprasījumu “Uzņemt attēlu” un attēlot uzņemto attēlu aktīvajā skatā.

Ievaddati: Redzot skatu “Darbības ar kameru”, lietotājs izvēlas uzņemt attēlu, un aktivizē pogu “Uzņemt attēlu”.

Darbības:

Sistēma nosūta pieprasījumu kamerai un gaida datus ar informāciju par uzņemto attēlu.

Izvaddati:

Sistēma, sagaidot datus par uzņemto attēlu, sekmīgā attēla uzņemšanas gadījumā aktīvajā skatā attēlo uzņemtās fotogrāfijas sīkattēlu, nesekmīgā gadījumā aktīvajā skatā attēlo kļūdas paziņojumu. Sistēma, sagaidot datus ar stāvokli “Nav savienojums ar kameru”, aktīvajā skatā attēlo kļūdas paziņojumu.

1.4. Ārējā saskarne

1.4.1. Lietotāja saskarne

Lietotāja saskarni veido pārlūkprogrammā atvērta spraudņa nodrošināta vietne – HTML dokuments. Grafiskā saskarne ir HTML dokuments attēlots pārlūkprogrammā. Grafiskie un interaktīvie elementi dalās grupās, kas tiek saukti par skatiem. Lietotājam vērot skatu nozīmē to, ka šobrīd ekrānā ir redzams specifisks komplekts ar saskarnes dizaina elementiem, ievadlaukiem, pogām un citiem HTML elementiem.

1.4.2. Aparatūras saskarne

Sistēma nodrošina darbu ar ārējo kameru (*Sony QX1*), kas ir nepieciešama sistēmas darbībai. Sistēmas programmatūras lietošanai ir nepieciešams strādājošs dators, kas aprīkots ar bezvadu tīkla adapteri, kas var pieslēgties kameras nodrošinātajam 802.11 b/g/n tīklam[2]

1.4.3. Programmatūras saskarne

Lai varētu lietot sistēmu, ir nepieciešams, lai uz lietotāja datora būtu uzstādīta tīmekļa lietojumprogrammatūra.

Sistēma paredz to, ka lietotāja datorā ir uzstādīta pārlūkprogramma (*Google Chrome*).

2. Programmatūras projektējuma apraksts

2.1. Ievads

Šī dokumenta nodaļa (turpmāk PPA) ir dokuments, kas balstīts uz LVS 72:1996 standartā aprakstīto ieteicamo praksi programmatūras projektējuma izstrādāšanai.

2.1.1. Projektējuma apraksta nolūks

Šī PPA nolūks ir kameras kontroles sistēmas un tās realizācijas plānojuma aprakstīšana. Ar šī dokumenta palīdzību tiks plānota sistēmas realizācija tā kā tā aprakstīta PPS uzskaitītajās funkcijās.

2.1.2. Darbības sfēra

Šī PPA darbības sfēra ir programmatūra, kura ir iepriekš aprakstīta PPS. Šo informāciju iespējams atrast PPS nodaļā "1.1.2. Darbības sfēra".

2.1.3. Saistība ar citiem dokumentiem

Šis dokuments ir saistīts ar sistēmas PPS.

2.2. Dekompozīcijas apraksts

Pamatojoties uz programmatūras prasību specifikāciju, tika konstatēts, ka programmatūras risinājuma izveidei ir nepieciešams izstrādāt atsevišķas komponentes, kas vienlaicīgi ir neatkarīgas, bet tām ir jāstrādā kopā, lai izpildītu prasības.

2.2.1. Grafiskā vide

Lietotājam būs nepieciešama grafiskā saskarne, kurā tas varēs redzēt savienojuma ar kameru stāvokli un sistēmas stāvokli. Tāpat grafiskajā saskarnē būs pieejami pogas u.c. interaktīvi rīki, kas tiks izmantoti, lai nodotu komandas kamerai. Grafiskajai saskarnei ir jābūt nodrošinātai lietotājam pierastā un ērtā veidā, tādēļ šī nolūka realizācijai tika izvēlēta pārlūkprogramma lietotāja datorā.

2.2.2. Spraudnis pārlūkprogrammā

Lai realizētu grafiskās saskarnes spēju sazināties ar fotokameru, ir nepieciešams starpnieks – spraudnis, kas tiek uzstādīts lietotāja datorā uzstādītajā pārlūkprogrammā.

Lai nodrošinātu stabilu grafiskās vides atbalstu uz populārākajām personālā datora operētājsistēmām, ir izvēlēta pārlūkprogramma *Google Chrome* (turpmāk pārlūks)[1]

Pārlūka dokumentācija nosaka tās spraudņu uzbūvi [3]. Spraudņa pakotne sastāv no vairākiem failiem: *HTML* dokumentiem, *CSS* stila lapas koda failiem, *JavaScript* koda failiem kā arī spraudņa pakas informatīvajiem konfigurācijas failiem.

Vadoties pēc specifikācijas, spraudņa realizēšanai būs nepieciešamas šādas lietas:

- Galvenais *HTML* dokuments (nodrošina informāciju grafiskā interfeisa ģenerēšanai, kā arī papildu resursu ielādes informāciju)
- Galvenais CSS stila lapa fails (nodrošina grafiskā interfeisa attēlojuma īpatnības)
- Galvenais JavaScript koda fails (uzglabā kodu, kas grafiskajā vidē ļauj interaktīvi reaģēt uz lietotāja ievadi un nodrošina saziņu ar citām sistēmām)
- Spraudņa informācijas fails “*manifest*” (uzglabā nepieciešamos datus spraudņa lietošanai pārlūkprogrammā, apraksta piekļuves atļaujas, ceļus uz fotoaparāta kontroles serveri)

Visi iepriekš minētie faili tiek sapakoti vienā arhīvā, kas kalpo kā neatkarīgs fails, kuru izmanto, lai uzstādītu spraudni lietotāja pārlūkā.

2.2.3. Fotoaparāta kontroles servera uzbūve

Tā kā spraudnis un tajā esošais kods nav spējīgs komunicēt ar tīkla ierīcēm, ir nepieciešams vēl viens starpnieks – kontroles serveris, kas pārveido komandas no spraudņa par komandām fotokamerai, kā arī nodrošinās komunikācijas procesu starp lietotāja datoru un fotokameru.

Fotoaparāta kontroles servera programmatūra (turpmāk – serveris) būs programmēta *Python* valodā, kas sastāvēs no vairākiem failiem, kas izvietoti vienā direktoriijā.

Lai nodrošinātu šī servera programmatūras darbību, ir nepieciešams izstrādāt šādus programmatūras moduļus:

- *chromeCommunication* modulis – atbild par saziņu starp pārlūkprogrammu un serveri, nosūta datus uz pārlūkprogrammatūru un nodrošina datu saņemšanu no pārlūka.
- *chromeMessageParser* modulis – apstrādā saņemtos datus un sagatavo pieprasījumus kamerai
- *cameraCommunication* modulis – atbild par saziņu starp serveri un kameru, izveido savienojumu ar kameru, sūta un saņem datus uz kameru, sagatavo no kameras saņemtos datus tālākai apstrādei *cameraResponseParser* modulim
- *cameraResponseParser* modulis – apstrādā no kameras saņemtos datus un sagatavo tos tālākai nodošanai *chromeCommunication* modulim

2.2.4. Fotoaparāta saskarne

Saziņu ar fotoaparātu nodrošina tiešais savienojums starp lietotāja datorā strādājošu bezvadu tīkla adapteri un fotoaparāta nodrošināto bezvadu tīklu. Saskarne tiek realizēta sūtot un saņemot speciālas komandas speciālā formātā (sīkāk skatīt nodaļā . 2.3.2)

2.2.5. Žurnālēšanas saskarnes modulis

Notikumu žurnāla sistēma ir nepieciešama, lai nodrošinātu iespēju nodot paziņojuma, kļūdas vai brīdinājuma datus sistēmas atklūdošanas vajadzībām. Žurnālēšanas dati tiek saglabāti īpašā formātā (sīkāk skatīt sadaļā 2.3.1) un saglabāti vēlākai apstrādei. Žurnālēšanas sistēma būs programmēta *Python* valodā.

2.3. Datu struktūras un formāti

2.3.1. Datu struktūra datu apmaiņai starp pārlūkprogrammatūru un serveri

Datu apmaiņai starp pārlūku un serveri notiek atbilstoši *Chrome Native Messaging* [9] protokolam, pārlūkā izmantojot *Native Messaging API*.

Ziņojumu nodošana notiek, izmantojot standartievadi (*stdin*) un standartizvadi (*stdout*). Ziņojumiem jābūt virknētiem (*serialized*) atbilstoši JSON datu formātam. Ziņojumiem ir jābūt kodētiem UTF-8 transformēšanas formātā. Katrs ziņojums sākas ar 32 bitu garu daļu sistēmai dabīgajā baitu secībā (*native byte order*), kas satur ziņojuma garumu. Viena ziņojuma maksimālais izmērs ir viens megabaits (1 MB).

Zemāk tiks aprakstīti JSON datu masīvi, kas tiek izmantoti ziņojumu apmaiņai starp pārlūku un serveri.

2.3.1.1 Komandas izpildes pieprasījums:

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|------------------------------|--------------------------------|------------------------------|--------------|-----------------------------------|
| type | <i>string</i> | type | Jā | Masīvu raksturojošā tipa atslēga. |
| type.command | <i>string</i> | Nav | Jā | Raksturo masīva tipu. |
| properties | <i>string</i> | <i>dict</i> tipa masīvs | Jā | Glabā datus par pieprasījumu. |
| properties.command | <i>string</i> | <i>string</i> | Jā | Pieprasījuma nosaukums |
| properties.command_parameter | <i>String</i> <i>boolean</i> | Nav | Nē | Pieprasījuma parametrs. |

Datu struktūras paraugs:

```
{
  "type": "command",
  "properties": {
    "command": "act_take_picture",
    "command_parameter": "none"
  }
}
```

2.3.1.2. Komandas izpildes atbilde:

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|-----------------------|---------------|------------------------------|--------------|--|
| type | <i>string</i> | command_response | Jā | Norāda masīva tipu. |
| type.command_response | <i>string</i> | Nav | Jā | Norāda, ka masīvs satur informāciju par kameras statusu. |
| properties | <i>string</i> | <i>dict</i> tipa masīvs | Jā | Satur datus par pieprasījumu |
| properties.command | <i>string</i> | Nav | Jā | Izpildītā pieprasījuma nosaukums |
| properties.response | <i>string</i> | Jā | Nē | Informācija par pieprasījuma izpildes rezultātu. |

Datu struktūras paraugs:

```
{
  "type": "command_response",
  "properties": {
    "command": "string",
    "response": "string"
  }
}
```

2.3.1.3. Kameras stāvokļa atbilde

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|---|---------------|--|--------------|--|
| type | <i>string</i> | camera_status | Jā | Norāda masīva tipu. |
| properties | <i>string</i> | <i>dict</i> tipa masīvs | Jā | Satur kameras stāvokļa atskaites datus |
| properties.description | <i>string</i> | current camera status | Nē | Masīva apraksts cilvēkam lasāmā formā |
| properties.status_data | <i>string</i> | <i>dict</i> ar kameras stāvokļa atskaiti | Jā | Satur kameras stāvokļa atskaiti |
| properties.status_data.current_camera_function | <i>string</i> | Nav | Jā | Kameras ieslēgtā funkcija |
| properties.status_data.exposure_compensation_step | <i>int</i> | Nav | Jā | Ekspozīcijas soļa parametrs |
| properties.status_data.current_focus_mode | <i>string</i> | Nav | Jā | Ieslēgtais fokusa režīms |

| | | | | |
|--|---------------|-----|----|--|
| properties.status_data. maximal_exposure_compensation | <i>int</i> | Nav | Jā | Ekspozīcijas parametra maksimālā vērtība |
| properties.status_data. minimal_exposure_compensation | <i>int</i> | Nav | Jā | Ekspozīcijas parametra minimālā vērtība |
| properties.status_ data.current_exposure_mode | <i>string</i> | Nav | Jā | Ieslēgtais ekspozīcijas režīms |
| properties.status_data. current_shoot_mode | <i>string</i> | Nav | Jā | Ieslēgtais uzņemšanas režīms |
| properties.status_data. battery_denominator | <i>int</i> | Nav | Jā | Baterijas maksimālā līmeņa parametrs |
| properties.status_data. current_battery_level | <i>int</i> | Nav | Jā | Baterijas atlikums |
| properties.status_data. still_size | <i>string</i> | Nav | Jā | Uzstādītā attēla uzņemšanas izšķirtspēja |
| properties.status_data. current_shutter_speed | <i>string</i> | Nav | Jā | Uzstādītais uzņemšanas slēdža ātrums |
| properties.status_data. storage_id | <i>string</i> | Nav | Jā | Atmiņas kartes nosaukums. Parametru var izmantot, lai secinātu, vai |

| | | | | |
|--|---------------|-----|----|---|
| | | | | ievietota atmiņas karte. |
| properties.status_data. current_flash_mode | <i>string</i> | Nav | Jā | Uzstādītais zibspuldzes režīms |
| properties.status_data. camera_status | <i>string</i> | Nav | Jā | Vai kamera šobrīd ir pieejama darbībām |
| properties.status_data. is_battery_near_empty | <i>string</i> | Nav | Jā | Vai baterija ir gandrīz tukša |
| properties.status_data. current_iso | <i>int</i> | Nav | Jā | Uzstādītais ISO parametrs |
| properties.status_data. current_white_balance_temperature | <i>int</i> | Nav | Jā | Uzstādītā baltā balansa vērtība |
| properties.status_data. current_white_balance_mode | <i>string</i> | Nav | Jā | Uzstādītais uzņemšanas baltā balansa režīms |
| properties.status_data. current_exposure_compensation | <i>int</i> | Nav | Jā | Uzstādītā uzņemšanas ekspozīcijas kompensācijas vērtība |
| properties.status_data. current_F_number | <i>string</i> | Nav | Jā | Aktuālā F skaitļa vērtība |
| properties.status_data. is_white_balance_available | <i>bool</i> | Nav | Jā | Vai ir iespējams mainīt baltā balansa režīmu |

| | | | | |
|--|---------------|-----|----|--|
| properties.status_data. current_still_quality | <i>string</i> | Nav | Jā | Attēla uzņemšanas formāts |
| properties.status_data. aspect_ratio | <i>string</i> | Nav | Jā | Attēla uzņemšanas malu attiecība |

Datu struktūras paraugs:

```
{
  "type": "camera_status",
  "properties": {
    "description": "current camera status",
    "status_data": {
      "current_camera_function": "Remote Shooting",
      "exposure_compensation_step": 1,
      "current_focus_mode": "MF",
      "maximal_exposure_compensation": 9,
      "current_exposure_mode": "Shutter",
      "current_shoot_mode": "Shutter",
      "battery_denominator": 4,
      "still_size": "20M",
      "current_battery_level": 4,
      "current_shutter_speed": "0.4\"",
      "storage_id": "Memory Card 1",
      "current_flash_mode": "off",
      "camera_status": "IDLE",
      "is_battery_near_empty": "",
      "current_iso": "100",
      "current_white_balance_temperature": -1,
      "current_white_balance_mode": "Auto WB",
      "current_exposure_compensation": 0,
      "current_F_number": "--",
      "is_white_balance_available": true,
      "current_still_quality": "RAW+JPEG",
      "aspect_ratio": "3:2",
      "minimal_exposure_compensation": -9
    }
  }
}
```

2.3.1.4. Sistēmas stāvokļa atbilde

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|------------------------|---------------|---|--------------|---|
| type | <i>string</i> | info | Jā | Masīva tips |
| properties | <i>string</i> | <i>dict</i> ar informatīvā ziņojuma parametriem | Jā | Glabā informatīvā paziņojuma datus |
| properties.status_code | <i>int</i> | Nav | Jā | Sistēmai paredzētais paziņojuma kods |
| properties.description | <i>string</i> | Nav | Jā | Informācijas paziņojums cilvēkam lasāmā formā |

Datu struktūras paraugs:

```
{
  "type": "system_status",
  "properties": {
    "status_code": int,
    "description": "string"
  }
}
```

2.3.1.5. Paziņojuma atbilde

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|------------------------|---------------|---|--------------|---|
| type | <i>string</i> | info | Jā | Masīva tips |
| properties | <i>string</i> | <i>dict</i> ar informatīvā ziņojuma parametriem | Jā | Glabā informatīvā paziņojuma datus |
| properties.info_code | <i>int</i> | Nav | Jā | Sistēmai paredzētais paziņojuma kods |
| properties.description | <i>string</i> | Nav | Jā | Informācijas paziņojums cilvēkam lasāmā formā |

Datu struktūras paraugs:

```
{
  "type": "info",
  "properties": {
    "info_code": 1,
    "description": "string"
  }
}
```

2.3.1.6. Brīdinājuma ziņojums

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|-------------------------|---------------|-----------------------------------|--------------|--|
| type | <i>string</i> | warning | Jā | Masīva tips |
| properties | <i>string</i> | <i>dict</i> ar brīdinājuma datiem | Jā | Glabā brīdinājuma datus |
| properties.warning_code | <i>int</i> | Nav | | Sistēmai paredzētais brīdinājuma kods |
| properties.description | <i>string</i> | Nav | Jā | Brīdinājuma apraksts cilvēkam lasāmā formā |

Datu struktūras paraugs:

```
{
  "type": "warning",
  "properties": {
    "warning_code": int,
    "description": "string"
  }
}
```

2.3.1.7. Kļūdas paziņojums

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|------------------------|---------------|------------------------------|--------------|---------------------------------------|
| type | <i>string</i> | error | Jā | Raksturo masīva tipu |
| properties | <i>string</i> | <i>dict</i> ar kļūdas datiem | Jā | Glabā kļūdas datus |
| properties.error_code | <i>int</i> | Nav | Jā | Sistēmai paredzētais kļūdas kods |
| properties.description | <i>string</i> | Nav | Jā | Kļūdas apraksts cilvēkam lasāmā formā |

Datu struktūras paraugs:

```
{
  "type" : "error",
  "properties": {
    "error_code": int,
    "description": "string"
  }
}
```

2.3.2. Datu struktūra datu apmaiņai starp serveri un kameru

Datu apmaiņa starp fotoaparātu un serveri notiek atbilstoši Sony Remote Camera API [10] noteiktajiem standartiem. Datu objekti tiek veidoti atbilstoši JSON-RPC [11] standartam.

2.3.2.1. Pieprasījuma kamerai paraugs:

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|-----------|---------------|------------------------------|--------------|--|
| method | <i>string</i> | Nav | Jā | Glabā datus par pieprasījuma veidu |
| params | <i>list</i> | Nav | Jā | Glabā pieprasījuma parametrus (var būt dažādu tipu atkarībā no pieprasījuma) |
| id | <i>int</i> | Nav | Jā | Pieprasījuma identifikators |
| version | <i>string</i> | Nav | Jā | Pieprasījuma versija (API viena tipa pieprasījumiem var būt vairākas versijas) |

Datu struktūras paraugs:

```
{  
  "method": "getEvent",  
  "params": [False],  
  "id": 1,  
  "version": "1.0"  
}
```

2.3.2.2. Atbilde pieprasījuma izpildes gadījumā

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|-----------|-------------|------------------------------|--------------|--|
| result | <i>list</i> | Nav | Jā | Satur sarakstu ar atbildes parametriem |
| id | <i>int</i> | Nav | Jā | Atbildes identifikators |

Datu struktūras piemērs:

```
{  
  "result": [0],  
  "id": 1  
}
```

2.3.2.3. Atbilde pieprasījuma neizpildīšanas gadījumā

| Nosaukums | Tips | Noklusētā atbilstošā vērtība | Vai obligāts | Komentārs |
|-----------|-------------|------------------------------|--------------|--|
| error | <i>list</i> | Nav | Jā | Satur kļūdas kodu un īsu kļūdas aprakstu |
| id | <i>int</i> | Nav | Jā | Kļūdas identifikators |

Datu struktūras piemērs:

```
{  
  "error": [5, "Illegal Request"],  
  "id": 1  
}
```

2.4 Moduļu projektējums

2.4.1 Pārlūka spraudnis

2.4.1.1. main.js

2.4.1.1.1 debugLog

| | |
|-------------------|---|
| Apraksts | Izvada notiekošos procesus žurnālēšanas skatam. |
| Ievades parametri | text |
| Apstrāde | Saņem ievades parametru un pievieno tam laika birku (<i>timestamp</i>). |
| Izvade | Nosūta izvadi uz <i>debug</i> lauku. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.2 appendMessage

| | |
|-------------------|--|
| Apraksts | Pievieno statusa lauka attēlojumam ziņojumus |
| Ievades parametri | message |
| Apstrāde | Saņem ievades parametru un veic attēlojumu atbilstoši ievades parametra tipam. |
| Izvade | Izvade uz statusa lauku. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.3 updateUiState

| | |
|-------------------|--|
| Apraksts | Atklāj vai slēpj interaktīvos elementus atkarībā no savienojuma ar serveri stāvokļa. |
| Ievades parametri | Nav. |
| Apstrāde | Pārbauda savienojuma ar serveri stāvokli, atjaunina skatu. |
| Izvade | Slēpj vai atklāj interaktīvos elementus |
| Kļūdas | Nav paredzētas. |

2.4.1.1.4 sendNativeMessage

| | |
|-------------------|--|
| Apraksts | Nosūta ziņojumu |
| Ievades parametri | message |
| Apstrāde | Nosūta ziņojumu serverim. |
| Izvade | Izvada ziņu par pieprasījuma veikšanu grafiskajā saskarnē. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.5 onNativeMessage

| | |
|-------------------|--|
| Apraksts | Veic izmaiņas skatā atkarībā no ienākošā parametra datiem. |
| Ievades parametri | message |
| Apstrāde | Pārbauda, kādi parametri ir ienākošajā parametrā. Atjaunina skatu “Darbības ar kameru”. |
| Izvade | Izmaiņas skatā. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.6 sendConnectMessage

| | |
|-------------------|--|
| Apraksts | Sagatavo un nosūta pieprasījumu serverim uzsākt savienojumu ar kameru. |
| Ievades parametri | Nav. |
| Apstrāde | Izveido pieprasījumu. Nosūta pieprasījumu. Izvada datus par pieprasījumu skatam “Darbības ar kameru” |
| Izvade | Izmaiņas skatā. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.7. sendParameterRequest

| | |
|-------------------|--|
| Apraksts | Sagatavo un nosūta pieprasījumu serverim iegūt kameras stāvokļa datus. |
| Ievades parametri | Nav. |
| Apstrāde | Izveido pieprasījumu. Nosūta pieprasījumu. Izvada datus par pieprasījumu skatam “Darbības ar kameru” |
| Izvade | Izmaiņas skatā. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.8. sendConnectionStatusRequest

| | |
|-------------------|--|
| Apraksts | Sagatavo un nosūta pieprasījumu serverim iegūt datus par savienojumu ar kameru. |
| Ievades parametri | Nav. |
| Apstrāde | Izveido pieprasījumu. Nosūta pieprasījumu. Izvada datus par pieprasījumu skatam “Darbības ar kameru” |
| Izvade | Izmaiņas skatā. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.9 shootImageNow

| | |
|-------------------|--|
| Apraksts | Sagatavo un nosūta pieprasījumu serverim pieprasīt kamerai attēla uzņemšanu. |
| Ievades parametri | Nav. |
| Apstrāde | Izveido pieprasījumu. Nosūta pieprasījumu. Izvada pieprasījuma datus žurnālēšanas funkcijai. |
| Izvade | Nav. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.10. onDisconnected

| | |
|-------------------|---|
| Apraksts | Izvada datus skatā servera darbības apstāšanās gadījumā. |
| Ievades parametri | Nav. |
| Apstrāde | Attēlo skatā datus par kļūdu. Maina skatu uz “Sākt darbu”. |
| Izvade | Izmaiņas skatā. |
| Kļūdas | Nav paredzētas. |

2.4.1.1.11 connect

| | |
|-------------------|---|
| Apraksts | Uzsāk servera darbu. Pēc servera darbības uzsākšanas maina skatu uz “Darbības ar kameru”. |
| Ievades parametri | Nav. |
| Apstrāde | Izvada skatā datus par servera sāknēšanu. Nodibina datu kanālu ar serveri. Nomaina skatu. |
| Izvade | Izmaiņas skatā. |
| Kļūdas | Nav paredzētas. |

2.4.2. Servera moduļi

2.4.2.1 chromeCommunication modulis

Atbild par saziņu starp pārlūkprogrammu un serveri, nosūta datus uz pārlūkprogrammatūru un nodrošina datu saņemšanu no pārlūka.

2.4.2.1.1 send_message

| | |
|------------------------|--|
| Apraksts | Nosūta ziņu pārlūka spraudnim. |
| Ievades parametri | message |
| Ievades parametru tips | <i>string</i> |
| Apstrāde | Nolasa ziņas garumu, konvertē to binārā formā. Nosūta ziņas garumu spraudnim. Nosūta ziņu spraudnim. |
| Izvade | Nosūta izvadi uz standartizvadi (<i>stdout</i>). |
| Kļūdas | Nav paredzētas. |

2.4.2.1.2 read_input

| | |
|------------------------|---|
| Apraksts | Saņem ziņu no pārlūka spraudņa. |
| Ievades parametri | message |
| Ievades parametru tips | <i>string</i> |
| Apstrāde | Ielasa saņemamās ziņas garumu, konvertē. Ielasa ziņu. Pārveido ziņu no binārā formāta par <i>string</i> . |
| Izvade | Atgriež konvertēto ziņu. |
| Kļūdas | Nav paredzētas. |

2.4.2.2 chromeMessageParser modulis

Apstrādā saņemtos datus un sagatavo pieprasījumus kamerai.

2.4.2.2.1 convert_message_to_json_object

| | |
|------------------------|---|
| Apraksts | Pārveidot ievadīto string tipa vērtību tālāk apstrādājamā <i>dict</i> objektā |
| Ievades parametri | message |
| Ievades parametru tips | <i>string</i> |
| Apstrāde | Konvertē ievadīto parametru. |
| Izvade | Atgriež dict kopu. |
| Kļūdas | Nav paredzētas. |

2.4.2.2.2 process_message_to_json_object

| | |
|------------------------|---|
| Apraksts | Apstrādā saņemto objektu nodošanai tālāk cameraCommunication modulim |
| Ievades parametri | message |
| Ievades parametru tips | <i>dict</i> |
| Apstrāde | Atlasa ievadītajā parametrā nepieciešamos laukus. Atlasītos laukus ievieto jaunā <i>dict</i> objektā nepieciešamajā formātā. |
| Izvade | Atgriež dict kopu. |
| Kļūdas | Ja ievadītajā parametrā ir nepietiekami vai kļūdaini ievades lauki, atgriež <i>dict</i> ar kļūdas paziņojumu. |

2.4.2.2.3 process_message_all

| | |
|------------------------|---|
| Apraksts | Apvieno iepriekšējās funkcijas vienotam izsaukumam |
| Ievades parametri | message |
| Ievades parametru tips | <i>string</i> |
| Apstrāde | Izsauc <code>convert_message_to_json_object</code> funkciju. Izsauc <code>process_message</code> funkciju. |
| Izvade | Atgriež dict kopu. |
| Kļūdas | Ja ievadītajā parametrā ir nepietiekami vai kļūdaini ievades lauki, atgriež <i>dict</i> ar kļūdas paziņojumu. |

2.4.2.3 CameraCommunication modulis

Atbild par saziņu starp serveri un kameru, izveido savienojumu ar kameru, sūta un saņem datus uz kameru, sagatavo no kameras saņemtos datus tālākai apstrādei `cameraResponseParser` modulim.

2.4.2.3.1 get_camera_xml_adress

| | |
|------------------------|---|
| Apraksts | Iegūt piekļuves adresi XML datnei, kas satur kameras API piekļuves ceļus. |
| Ievades parametri | Nav |
| Ievades parametru tips | - |
| Apstrāde | Atrod kameru tīklā. Iegūst kameras <i>broadcast</i> ziņu. Izdala XML datnes piekļuves saiti no iegūtajiem datiem. |
| Izvade | Saglabā piekļuves saiti. |
| Kļūdas | Savienojuma ieguves neizdošanās gadījumā atgriež <i>dict</i> ar kļūdas aprakstu. |

2.3.2.3.2 fetch_camera_access_urls

| | |
|------------------------|--|
| Apraksts | Iegūt kameras API piekļuves ceļus |
| Ievades parametri | Nav. |
| Ievades parametru tips | - |
| Apstrāde | Ielasa XML datni. Apstrādā XML datni, iegūst API piekļuves adreses. Saglabā API piekļuves adreses. |
| Izvade | Nav. |
| Kļūdas | Ielasīšanas neizdošanās gadījumā atgriež <i>dict</i> ar kļūdas aprakstu. |

2.3.2.3.3 check_if_camera_is_connected

| | |
|------------------------|--|
| Apraksts | Pārbauda, kāds ir atmiņā saglabātais savienojuma ar kameru . |
| Ievades parametri | Nav |
| Ievades parametru tips | - |
| Apstrāde | Pārbauda kameras savienojuma stāvokli atmiņā. |
| Izvade | Atgriež stāvokļa parametru. |
| Kļūdas | Ielasīšanas neizdošanās gadījumā atgriež <i>dict</i> ar kļūdas aprakstu. |

2.3.2.3.4 convert_request_parameter_boolean_values

| | |
|------------------------|--|
| Apraksts | Pārveido pieprasījuma parametrus par būleānām |
| Ievades parametri | request_parameters |
| Ievades parametru tips | <i>list</i> ar dažādu tipu vērtībām. |
| Apstrāde | Sagatavo izvades parametrus. Ja satur <i>string</i> vērtību “true” “false”, pārveido tās par atbilstošām būleānām. |
| Izvade | <i>list</i> ar konvertētajiem parametriem, |
| Kļūdas | Nav paredzētas |
| | |

2.3.2.3.5 evaluate_request_success

| | |
|------------------------|--|
| Apraksts | Pārbauda, vai pieprasījumu kamerai ir izdevies izpildīt. |
| Ievades parametri | response_input |
| Ievades parametru tips | <i>dict</i> |
| Apstrāde | Pārbauda, vai ievades parametrā ir atslēga “error”, atgriež būleānu. |
| Izvade | Atgriež pārbaudes rezultātu. |
| Kļūdas | Nav paredzētas. |

2.3.2.3.6 send_request_to_camera

| | |
|------------------------|--|
| Apraksts | Veic pieprasījumu kamerai, saņem pieprasījuma atbildi, konvertē atbildi apstrādājamā formātā. |
| Ievades parametri | input_param, method, address, version |
| Ievades parametru tips | <i>string, list, string, string</i> |
| Apstrāde | Izveido pieprasījumu atbilstošajā formā. Nosūta pieprasījumu. Iegūst pieprasījuma atbildi. Konvertē pieprasījuma atbildi. |
| Izvade | Atgriež <i>dict</i> ar atbildes datiem. |
| Kļūdas | Savienojuma neizdošanās gadījumā atgriež <i>dict</i> ar kļūdas datiem. |

2.3.2.3.7 get_current_state

| | |
|------------------------|---|
| Apraksts | Veic pieprasījumu saņemt kameras aktuālos darbības parametrus |
| Ievades parametri | instant |
| Ievades parametru tips | <i>bool</i> |
| Apstrāde | Sagatavo <i>instant</i> parametru, ja tāds nav padots. Izsauc <code>send_request_to_camera</code> , params padodot <i>instant</i> parametru. |
| Izvade | Atgriež pieprasījuma rezultātu. |
| Kļūdas | Nav paredzētas. |

2.3.2.3.8 act_take_picture

| | |
|------------------------|--|
| Apraksts | Nodod pieprasījumu uzņemt |
| Ievades parametri | Nav. |
| Ievades parametru tips | - |
| Apstrāde | Izsauc send_request_to_camera ar atbilstošajiem parametriem. |
| Izvade | Atgriež pieprasījuma rezultātu. |
| Kļūdas | Nav paredzētas. |

2.3.2.3.9 connect_to_camera

| | |
|------------------------|--|
| Apraksts | Izveido sākotnējo savienojumu ar kameru. |
| Ievades parametri | Nav. |
| Ievades parametru tips | - |
| Apstrāde | Izsauc get_camera_xml_address. Izsauc fetch_camera_access_urls. |
| Izvade | Atgriež atmiņā saglabāto kameras savienojuma stāvokli. |
| Kļūdas | Nav paredzētas. |

2.3.2.3.10 perform_camera_action

| | |
|------------------------|---|
| Apraksts | Apstrādā padoto <i>dict</i> un veic izsaukumu uz atbilstošo funkciju, atgriež darbības rezultātu. |
| Ievades parametri | instance_name, command_message |
| Ievades parametru tips | <i>string, dict</i> |
| Apstrāde | Atlasa no command_message izsaucamo komandu un tās parametrus (ja tādi ir). Izsauc nepieciešamo komandu. Sagatavo atgriežamos parametrus. |
| Izvade | <i>dict</i> ar izsuktās komandas nosaukumu un atbildes datiem. |
| Kļūdas | Nav paredzētas. |

2.4.2.4 cameraResponseParser modulis

Apstrādā no kameras saņemtus datus un sagatavo tos tālākai nodošanai chromeCommunication modulim.

2.4.2.4.1 parse_camera_state_response

| | |
|------------------------|--|
| Apraksts | Apstrādā saņemtus kameras stāvokļa datus, atlasa nepieciešamos un izveido atbilstošu <i>dict</i> . |
| Ievades parametri | in_parameters |
| Ievades parametru tips | <i>dict</i> |
| Apstrāde | Atlasa no in_parameters nepieciešamos datus. Sagatavo atbildes ziņojumu atbilstošajā formā. |
| Izvade | <i>dict</i> ar atbildes datiem. |
| Kļūdas | Kameras pieprasījuma atbildei saturot datus par kļūdu, atgriež kļūdas paziņojumu. |

2.4.2.4.2 parse_init_connection_dict

| | |
|------------------------|--|
| Apraksts | Apstrādā saņemtus sākotējā savienojuma izveides pieprasījuma rezultātu, izveido atbilstošu <i>dict</i> . |
| Ievades parametri | in_parameters |
| Ievades parametru tips | <i>dict</i> |
| Apstrāde | Apstrādā saņemtus datus, sagatavo ziņojumu par savienojuma stāvokli. |
| Izvade | <i>dict</i> ar savienojuma stāvokļa ziņojumu. |
| Kļūdas | Ja in_parameters satur kļūdas paziņojumu, to nodod tālāk neizmainītu. |

2.4.2.4.3 parse_act_take_picture_dict

| | |
|------------------------|--|
| Apraksts | Apstrādā saņemtos attēla uzņemšanas pieprasījuma rezultātu, izveido atbilstošu <i>dict</i> |
| Ievades parametri | <i>in_parameters</i> |
| Ievades parametru tips | <i>dict</i> |
| Apstrāde | Apstrādā saņemtos datus, sagatavo ziņojumu par savienojuma stāvokli. |
| Izvade | <i>dict</i> ar savienojuma stāvokļa ziņojumu. |
| Kļūdas | Nav paredzētas. |

2.4.2.4.4 parse_camera_response

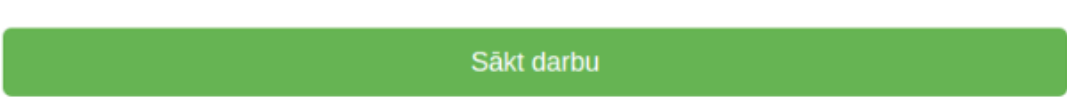
| | |
|------------------------|--|
| Apraksts | Atlasa no ievaddatiem izpildītās komandas vērtību, nodod datus apstrādei atbilstošajai apstrādes funkcijai, konvertē rezultāta datus uz <i>string</i> . |
| Ievades parametri | <i>in_parameters</i> |
| Ievades parametru tips | <i>dict</i> |
| Apstrāde | Atkarībā no <i>in_parameters</i> pieejamā izsauktās komandas parametra izsauc atbilstošu apstrādes funkciju. Konvertē atgriezto funkcijas rezultātu par <i>string</i> . |
| Izvade | Atgriež darbības rezultātu. |
| Kļūdas | Nav paredzētas. |

2.5. Lietotāja saskarnes apraksts

Lietotāja saskarnes aprakstā tiks apskatīti skati “Sākt darbu” un “Darbības ar kameru”.

2.5.1 Skats “Sākt darbu”

FOTOAPP kontrolē savu kameru.

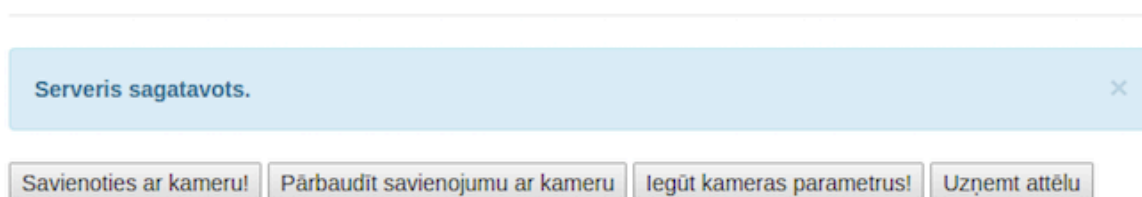


Sākt darbu

Skatu izmanto, lai aktivizētu sistēmas darbību, lietotājam izmantojot pogu “Sākt darbu”, tiek palaists Python serveris un notiek pāreja uz skatu “Darbības ar kameru”.

2.5.2 Skats “Darbības ar kameru”

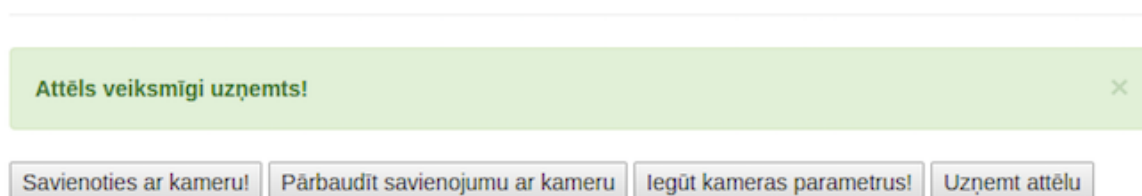
FOTOAPP kontrolē savu kameru.



Sākot darbu šai skatā, lietotājs tiek informēts par servera darbības stāvokli. Lietotājam šajā skatā ir iespējams izsaukt šādu funkcionalitāti:

- Izveidot savienojumu ar kameru
- Pārbaudīt savienojuma ar kameru stāvokli
- Iegūt kameras aktuālos darbības parametrus
- Uzņemt attēlu.

FOTOAPP kontrolē savu kameru.



Pēc attēla uzņemšanas pieprasījuma lietotājam tiek izvadīts paziņojums par pieprasījuma rezultātu (attēls uzņemts vai arī attēla uzņemšanas pieprasījumu neizdevās izpildīt).

FOTOAPP kontrolē savu kameru.

Ir savienojums ar kameru.



Savienoties ar kameru!

Pārbaudīt savienojumu ar kameru

Iegūt kameras parametrus!

Uzņemt attēlu

| Parametrs | Vērtība |
|-------------------------------|-----------------|
| current_camera_function | Remote Shooting |
| exposure_compensation_step | 1 |
| current_focus_mode | MF |
| maximal_exposure_compensation | 9 |
| current_exposure_mode | Shutter |
| current_shoot_mode | Shutter |
| battery_denominator | 4 |
| still_size | 20M |
| current_battery_level | 1 |
| current_shutter_speed | 0.4" |
| storage_id | Memory Card 1 |

Pēc parametru pieprasījuma lietotājam tiek izvadīti aktuālie kameras uzņemšanas parametri, vai atgriezts paziņojums par kļūdu pieprasījuma izpildē.

3. Testēšanas dokumentācija

Šajā daļā aprakstīta projekta testēšanas dokumentācija. Testēšana tika veikta katrai no izstrādātajām funkcijām, izmantojot t.s. “melnās kastes” principu, t.i. programmatūra tiek darbināta ar testiem, pieņemot, ka programmatūras izejas kods nav pieejams.

Funkciju testēšana notiek paralēli funkciju izstrādei.

3.1. ChromeCommunication moduļa testi

3.1.1. send_message funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|------------------------------|--|-----------------------|
| 1. | Tiek nodoti ievades dati, sagatavoti identiski pareizi dati formāta salīdzināšanai un salīdzināts funkcijas rezultāts ar paredzēto. | <i>String</i> tipa ievaddati | Ievadītie dati tiek sagatavoti nosūtīšanai pārlūka spraudnim atbilstošajā formātā (sīkāk skatīt 2.3.1) un nosūtīti pārlūkam. | Atbilst sagaidāmajam. |
| 2. | Tiek nodoti ievades dati un pārbaudīts vai izvades datu tips atbilst nepieciešamajam. | <i>String</i> tipa ievaddati | Ievaddati ir atbilstošā formātā (sīkāk skatīt 2.3.1). | Atbilst sagaidāmajam. |

3.1.2. read_input funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|------------------------------|--|-----------------------|
| 1. | Tiek nodoti ievades dati, sagatavoti identiski pareizi dati formāta salīdzināšanai un salīdzināts funkcijas rezultāts ar paredzēto. | <i>String</i> tipa ievaddati | Ievadītie dati tiek sagatavoti nosūtīšanai pārlūka spraudnim | Atbilst sagaidāmajam. |

| | | | | |
|----|---|------------------------------|---|-----------------------|
| | | | atbilstošajā formātā (sīkāk skatīt 2.3.1) un nosūtīti pārlūkam. | |
| 2. | Tiek nodoti ievades dati un pārbaudīts vai izvades datu tips atbilst nepieciešamajam. | <i>String</i> tipa ievaddati | Ievaddati ir atbilstošā formātā (sīkāk skatīt 2.3.1). | Atbilst sagaidāmajam. |

3.1.3. process_message funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|---|---|------------------------|
| 1. | Tiek pārbaudīta funkcijas izvaddatu pareizība, padodot komandu ar <i>string</i> tipa parametru. | <i>dict</i> ar aizpildītiem “command” un “command_params” laukiem (abi lauki <i>string</i> tipa). | Tiek sagatavots atbilstošs izvades objekts ar pareiziem komandas un komandas parametru laukiem. | Atbilst sagaidāmajam.. |
| 2. | Tiek pārbaudīta funkcijas izvaddatu pareizība, padodot komandu ar <i>dict</i> tipa parametru. | <i>dict</i> ar aizpildītiem “command” (ar <i>string</i>) un “command_params” (ar <i>string</i> vērtību pāri aizpildīts <i>dict</i>) laukiem. | Tiek sagatavots atbilstošs izvades objekts ar pareiziem komandas un komandas parametru laukiem. | Atbilst sagaidāmajam. |
| 3. | Tiek pārbaudīta funkcijas izvaddatu pareizība, padodot komandu bez komandas parametriem. | <i>Dict</i> ar aizpildītu “command” lauku un tukšu “command_params” lauku | Tiek sagatavots atbilstošs izvades objekts ar pareizu komandas un | Atbilst sagaidāmajam. |

| | | | | |
|----|--|---|---|-----------------------|
| | | | tukšu komandas parametru lauku. | |
| 4. | Tiek pārbaudīta funkcijas darbība, padodot ievaddatus bez funkcijas komandas un komandas parametru datiem. | <i>Dict</i> bez “command” un “command_params” laukiem | Tiek atgriezts izvades objekts ar kļūdas paziņojumu | Atbilst sagaidāmajam. |

3.1.4. process_message_all funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|----------------------------|---|--------------------------------------|
| 1. | Tiek pārbaudīta funkcijas darbība, padodot ievaddatus ar aizpildītiem komandas un komandas parametru laukiem | <i>string</i> tipa vērtība | Tiek sagatavots atbilstošs izvades objekts ar pareiziem komandas un komandas parametru laukiem. | Atbilst sagaidāmajam. |
| 2. | Tiek pārbaudīta funkcijas darbība, padodot komandu bez komandas parametriem. | <i>string</i> tipa vērtība | Tiek sagatavots atbilstošs izvades objekts ar pareizu komandas un tukšu komandas parametru lauku. | Atbilst sagaidāmajam. |
| 3. | Tiek pārbaudīta funkcijas darbība, padodot datus bez komandas un komandas parametru laukiem | <i>string</i> tipa vērtība | Tiek atgriezts izvades objekts ar kļūdas paziņojumu | Kļūda objekta apstrādē. Izlabots. |

3.2. cameraCommunication moduļa testi

3.2.1. get_camera_xml_address funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|----------------|-----------|-----------------------|-----------------|
| | | | | |

| | | | | |
|----|--|-----|---------------------------|-----------------------|
| 1. | Tiek mēģināts izveidot savienojumu ar kameru un iegūt XML faila adresi ar izveidotu tīkla savienojumu | Nav | Iegūst adresi | Atbilst sagaidāmajam. |
| 2. | Tiek mēģināts izveidot savienojumu ar kameru un iegūt XML faila adresi bez izveidotu tīkla savienojumu | Nav | Atgriež kļūdas paziņojumu | Atbilst sagaidāmajam. |

3.2.2. fetch_camera_access_urls funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|-----------|---------------------------|-----------------------|
| 1. | Tiek pārbaudīts, vai kamera atgriež kļūdas paziņojumu, ja ir zaudēts savienojums / nav iespējams iegūt piekļuves datus | Nav | Atgriež kļūdas paziņojumu | Atbilst sagaidāmajam. |
| 2. | Pārbaude, vai ir pareizi uzstādīta kameras uzņemšanas API piekļuves saite | Nav | Uzstāda parametru. | Atbilst sagaidāmajam. |
| 3. | Pārbaude, vai ir pareizi uzstādīta kameras satura API piekļuves saite | Nav. | Uzstāda parametru. | Atbilst sagaidāmajam. |
| 4. | Pārbaude, vai ir pareizi uzstādīta kameras sistēmas API piekļuves saite | Nav. | Uzstāda parametru. | Atbilst sagaidāmajam. |

3.2.4. is_camera_connected funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|-----------|-----------------------|-----------------------|
| 1. | Pārbauda, vai iekšējais lauks, kas uztur savienojuma | Nav | Nav | Atbilst sagaidāmajam. |

| | | | | |
|----|---|-----|---|-----------------------|
| | stāvokli pēc inicializācijas ir pareizā stāvoklī | | | |
| 2. | Pārbauda, vai ar eksistējošu savienojumu funkcija atgriež pareizus datus par savienojuma stāvokli | Nav | Atgriež pareizus datus par savienojuma stāvokli | Atbilst sagaidāmajam. |
| 3. | Pārbauda, vai bez savienojuma ar kameru funkcija atgriež pareizus datus par savienojuma stāvokli | Nav | Atgriež pareizus datus par savienojuma stāvokli | Atbilst sagaidāmajam. |

3.2.5. convert_parameter_request_boolean_values funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|---|--------------------------------|-----------------------|
| 1. | Pārbauda, vai ar ievadītiem parametriem bez būleānām atgriež pareizu rezultātu | List bez būleānas tipa parametriem. | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai ievadītiem parametriem ar <i>True</i> tipa būleānām atgriež pareizu rezultātu | List ar <i>True</i> tipa būleānas vērtību. | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |
| 3. | Pārbauda, vai ievadītiem parametriem ar <i>False</i> tipa būleānām atgriež pareizu rezultātu | List ar <i>False</i> tipa būleānas vērtību. | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |
| 4. | Pārbauda, vai ievadītiem parametriem ar jaukta tipa vērtībām atgriež pareizu rezultātu | List ar jaukta tipa vērtībām | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |
| 5. | Pārbauda, vai ievadītiem parametriem ar tukšu sarakstu | Tukšs list | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |

evaluate_request_success funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|-------------------------------------|--------------------------------|-----------------------|
| 1. | Pārbauda, vai ar ievades datiem, kas satur datus par izpildes kļūdu, tiek atgriezta korekta vērtība | <i>dict</i> ar atbilstošām vērtībām | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai ar ievades datiem, kas satur datus par veiksmīgu pieprasījuma izpildi, tiek atgriezta korekta vērtība | <i>dict</i> ar atbilstošām vērtībām | Atgriež korektus izvades datus | Atbilst sagaidāmajam. |

3.2.7. send_request_to_camera funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|--|--|-----------------------|
| 1. | Pārbauda, vai kameras pieprasījums izpildas veiksmīgi (iepriekš ir izveidots savienojums). | Tekošo parametru saņemšanai nepieciešamie parametri. | Funkcija atgriež pieprasījuma datus. | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai tiek atgriezta kļūdas paziņojums, ja tiek mēģināts izdarīt pieprasījumu kamerai un nav izveidots savienojums ar kameru. | Tekošo parametru saņemšanai nepieciešamie parametri. | Funkcija atgriež atbildi, kas satur datus par kļūdu. | Atbilst sagaidāmajam. |

3.2.8. connect_to_camera funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|-----------|-----------------------------------|-----------------------|
| 1. | Mēģinājums izveidot savienojumu ar kameru, ja ir | Nav | Atgriež korektu būvēšanas vērtību | Atbilst sagaidāmajam. |

| | | | | |
|----|---|-----|---|-----------------------|
| | izveidots tīkla savienojums ar kameru | | par savienojuma stāvokli | |
| 2. | Mēģinājums izveidot savienojumu ar kameru, ja nav izveidots tīkla savienojums ar kameru | Nav | Atgriež korektu būleānas vērtību par savienojuma stāvokli | Atbilst sagaidāmajam. |

3.2.9. get_current_state funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|---|-----------------------------------|---------------------------------------|
| 1. | Pārbauda, vai funkcijas atgriezto datu tips ir korekts | Nav | Datu tips korekts (<i>dict</i>) | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai ir iespējams padot nepareizu datu tipu parametram <i>instant</i> | Parametrs instant ar <i>string</i> tipu | | Kļūda pieprasījuma izpildē. Izlabots. |

3.3. cameraResponseParser moduļa testi

3.3.1. parse_current_state_response funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|--|---------------------------------|-----------------------|
| 1. | Pārbauda, vai funkcija korekti atlasa datus izvadei ar veiksmīga pieprasījuma atbildi. | <i>dict</i> ar komandas nosaukumu un stāvokļa datiem | Korekti apstrādāti izvades dati | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai funkcija korekti atlasa datus izvadei ar neveiksmīga pieprasījuma atbildi. | <i>dict</i> ar komandas nosaukumu un stāvokļa datiem | Korekti apstrādāti izvades dati | Atbilst sagaidāmajam. |

3.3.2. parse_connection_status_request funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|---|----------------------------------|--|
| 1. | Pārbauda, vai funkcija korekti atgriež datus ar ievades datiem par esošu savienojumu | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai funkcija korekti atgriež datus ar ievades datiem par savienojuma neesamību. | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam.. |
| 3. | Pārbauda, vai funkcija korekti atgriež datus ar ziņojumu par kļūdu. | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam.. |
| 4. | Pārbauda, vai funkcija korekti darbojas ar bojātiem ieejas datiem. | <i>Dict</i> trūkstošiem datiem. | Atgriež kļūdas paziņojumu. | Kļūda pieprasījuma izpildē. Izlabots. |

3.3.3. parse_init_connection_request funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|---|---|----------------------------------|-----------------------|
| 1. | Pārbauda, vai funkcija korekti atgriež datus ar ievades datiem par esošu savienojumu | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai funkcija korekti atgriež datus ar ievades datiem par savienojuma neesamību. | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam. |
| 3. | Pārbauda, vai funkcija korekti atgriež datus ar ziņojumu par kļūdu. | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam. |

3.3.4. act_take_picture funkcijas testi

| Nr. | Testa apraksts | Ievaddati | Sagaidāmais rezultāts | Testa rezultāts |
|-----|--|---|----------------------------------|-----------------------|
| 1. | Pārbauda, vai funkcija korekti atgriež datus ar ievades datiem par veiksmīgi izpildītu pieprasījumu. | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem. | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam. |
| 2. | Pārbauda, vai funkcija korekti atgriež datus ar ievades datiem par neveiksmīgu pieprasījuma izpildi. | <i>Dict</i> ar komandas nosaukumu un stāvokļa datiem | Korekti apstrādāti izvades dati. | Atbilst sagaidāmajam. |

4. Projekta organizācija

Pirms sistēmas izstrādes tika noteiktas sistēmas izpildāmās prasības, kas jāsasniedz sistēmas izstrādes beigās. Pēc tam tika definētas metodes, ar kurām šo risinājumu sasniegt. Projekta izstrādes laikā izstrādātājs regulāri veica konsultācijas ar darba vadītāju, veicot viedokļu apmaiņu par projekta moduļu izstrādi, saņemot priekšlikumus turpmākajai projekta izstrādei. Projekta izstrādē bija šādi posmi:

- Vēlamo lietotāja scenāriju apkopošana, prasību uzskaitē un ierobežojumu noteikšana
- Sistēmas koncepcijas izveide
- Sistēmas prasību specifikācijas izveide
- Sistēmas projektējuma izstrāde
- Sistēmas koda izstrāde
- Sistēmas testēšana, kļūdu labojumu ieviešana

Projekta tehniskā realizācija sastāv no tīmekļa tehnoloģijās balstīta pārlūka spraudņa, kura darbības nodrošināšanai ir nepieciešams serveris, kas veic tiešo komunikāciju ar kameru.

Pārlūka spraudņa izstrādē tika izmantota programmēšanas valoda JavaScript, kā arī iezīmēšanas valoda HTML. Servera izstrādē tika izmantota programmēšanas valodas Python (versija 2.7.9).

Projekts izstrādāts patstāvīgi, konsultējoties ar darba vadītāju. Tāpat darba izstrādes ietvaros dažādu tehnisku risinājumu atrisinājumam tika izmantoti izstrādātāju kopienu forumi.

5. Konfigurācijas pārvaldība

Lai nodrošinātu projekta konfigurācijas pārvaldību, tika izmantota versiju kontroles sistēma *git* un projekta versiju glabāšanai tika izmantota datu krātuve, ko nodrošina <http://www.github.com>.

Versiju kontroles sistēma tika lietota, lai nepieciešamības gadījumā būtu iespējams atgriezties pie iepriekšējas sistēmas versijas, kas gan projekta izstrādes laikā nebija nepieciešams. Jauni nodevumi sistēmai tika veikti pēc katras būtiskas moduļu izmaiņas un virspusējas testēšanas, lai pārliecinātos par veikto izmaiņu darbību un saderību.

6. Kvalitātes nodrošināšana

Lai nodrošinātu programmatūras produkta kvalitāti, vispirms tika izstrādāta PPS. Pēc PPS saskaņošanas ar darba vadītāju, balsoties uz PPS, tika izstrādāts PPA.

PPA izstrādes laikā tika veiktas konsultācijas ar darba vadītāju. Programmatūras produkts tika izstrādāts, vadoties pēc PPA.

Programmatūras pirmkoda kvalitātes nodrošināšanai servera programmatūras izstrādē tika ievērots PEP8 standarts[8] labās programmēšanas stilam, kas atvieglo koda lasāmību, uztveramību un modificējamību. Atbilstoši šim standartam ir veidoti arī pirmkoda komentāri, lai atvieglotu koda apgūšanu cilvēkam, kas ar to līdz šim nav iepazinies.

Darba izstrādes laikā tika veikta regulāra vienībtestēšana izstrādās programmatūras moduļiem un labotas testēšanas laikā atklātās kļūdas un nepilnības. Testi un to rezultāti tika fiksēti testēšanas žurnālā.

Lai nodrošinātu versiju kontroli, tika izmantota versiju pārvaldības sistēma *git* [6] un izmantota datu krātuve github.com[7] versiju uzglabāšanai.

7. Darbietilpības novērtējums

7.1. COCOMO aprēķina metode

Darbietilpības novērtējumam izmantojot COCOMO metodi, tika iegūti šādi rezultāti:

| | |
|--|-------------------------|
| Type | SLOC |
| SLOC | 634 |
| Language type | 3rd Generation Language |
| Precedentness | Nominal |
| Development flexibility | Nominal |
| Required Software Reliability | Nominal |
| Data Base Size | Nominal |
| Product Complexity | Nominal |
| Developed for Reusability | Nominal |
| Documentation Match to Lifecycle Needs | Nominal |
| Architecture / Risk Resolution | Nominal |
| Team Cohesion | Nominal |
| Analyst Capability | Nominal |
| Programmer Capability | Low |
| Personnel Continuity | Low |
| Application Experience | Nominal |
| Platform Experience | Nominal |
| Language and Toolset Experience | Low |
| Process Maturity | Nominal |
| Time Constraint | Nominal |
| Storage Constraint | Nominal |
| Platform Volatility | Nominal |
| Use of Software Tools | Nominal |
| Multisite Development | Nominal |
| Required Development Schedule | Nominal |
| Effort | 2.7 |
| Schedule | 5.1 |

Pēc COCOMO aprēķina paredzētā darbietilpība projektam ir 2.7 personmēneši.

7.2. Faktiskā laika patēriņš

| Darbības kategorija | Darbības nozare | Tips | Datu formāts | Apraksts | Aptuvenais laika patēriņš |
|---------------------|-----------------|-------------|--------------|--|---------------------------|
| Spraudnis | Izpēte | | | Chrome spraudņu darbības un struktūras izpēte | 4:00:00 |
| Spraudnis | Izpēte | | | Parauga spraudņu meklēšana un izpēte | 4:00:00 |
| Spraudnis | Izpēte | | | Spraudņa struktūras prototipēšana | 3:00:00 |
| Spraudnis | Izpēte | | | Spraudņa testēšana | 3:00:00 |
| Spraudnis | | | JSON | Spraudņa struktūras sagatavošana un programmēšana | 0:30:00 |
| Klients | Dizains | | | Grafiskā dizaina projektēšana | 1:00:00 |
| Klients | Web Aplikācija | Skati | HTML | Aplikācijas pamatnes šablons | 0:15:00 |
| Klients | Web Aplikācija | Skati | CSS | Aplikācijas kopējā grafiskā stila skripta izstrāde | 0:15:00 |
| | | | | JavaScript programmēšanas pamatu apguve | 2:00:00 |
| | | | | Aplikācijas galvenais skripta arhitektūras plānošana | 2:00:00 |
| Klients | Web Aplikācija | | JavaScript | Aplikācijas galvenais skripta prototipēšana | 4:00:00 |
| Klients | Web Aplikācija | Kontrolieri | JavaScript | Aplikācijas galvenais skripts | 1:00:00 |
| | | | | Komunikācijas starp spraudni un serveri izpēte | 2:00:00 |
| Klients | Web Aplikācija | | JavaScript | Komunikācijas starp spraudni un | 2:00:00 |

| | | | | | |
|----------|-------------------|-------------|------------|--|---------|
| | | | | serveri prototipēšana | |
| Klients | Web Aplikācija | Modeļi | JavaScript | Servera startēšanas funkcionāls | 0:15:00 |
| Serveris | Ielādes sistēma | | Python | Servera ielādes funkcionāls | 1:00:00 |
| Klients | | Modeļi | JavaScript | Komunikācija ar serveri | 0:15:00 |
| Serveris | Testēšana | | | Python scriptu testēšanas principu izpēte | 2:00:00 |
| Serveris | Testēšana | | Python | Komunikācija ar spraudņa testu izstrāde | 0:15:00 |
| Serveris | | Modeļi | Python | Komunikācija ar spraudni | 1:00:00 |
| Serveris | | Modeļi | Python | Kļūdu formatēšanas standarta plānošana | 0:15:00 |
| Serveris | | Modeļi | Python | Atbilžu formatēšanas standarta plānošana | 0:15:00 |
| Serveris | Testēšana | | Python | Darbības "Savienoties ar kameru" testēšana | 0:30:00 |
| Serveris | | Kontrolieri | Python | Darbība "Savienoties ar kameru" | 1:00:00 |
| Klients | | Kontrolieri | JavaScript | Skats "Savienoties ar kameru" | 2:00:00 |
| Klients | Web Aplikācija | Skati | HTML | Skata "Savienoties ar kameru" šablons | 1:00:00 |
| Klients | Web Aplikācija | Skati | CSS | Skata "Savienoties ar kameru" stils | 1:00:00 |
| | | | | Kameras komunikācijas dokumentācijas izpēte | 8:00:00 |

| | | | | | |
|----------|-------------------|-------------|------------|---|----------|
| Klients | Web Aplikācija | | Python | Komunikācijas ar kameru prototipēšana | 1:00:00 |
| Serveris | Testēšana | | Python | Komunikācijas ar kameru skriptu testēšana | 0:30:00 |
| Serveris | | Modeļi | Python | Komunikācija ar kameru | 6:00:00 |
| | | | | Kameras datu dokumentācijas izpēte | 8:00:00 |
| Serveris | | Modeļi | Python | Kameras datu apstrādes prototipēšana | 8:00:00 |
| Serveris | Testēšana | | Python | Kameras datu apstrādes testēšana | 1:00:00 |
| Serveris | | Modeļi | Python | Kameras datu apstrāde | 8:00:00 |
| Serveris | Testēšana | | Python | Kameras datu derīguma pārbaudes testēšana | 0:15:00 |
| Serveris | Testēšana | | Python | Darbības "Iegūt kameru stāvokļa atskaiti" testēšana | 1:30:00 |
| Serveris | | Kontrolieri | Python | Darbība "Iegūt kameru stāvokļa atskaiti" | 2:00:00 |
| Serveris | Testēšana | | Python | Darbības "Uzņemt attēlu" testēšana | 0:30:00 |
| Serveris | | Kontrolieri | Python | Darbība "Uzņemt attēlu" | 0:15:00 |
| Klients | Web Aplikācija | Kontrolieri | JavaScript | Skats "Darbības ar kameru" | 2:00:00 |
| Klients | Web Aplikācija | Skati | HTML | Skata "Darbības ar kameru" šablons | 2:00:00 |
| Klients | Web Aplikācija | Skati | CSS | Skata "Darbības ar kameru" stils | 1:00:00 |
| | | | | Specifikācijas izstrāde | 40:00:00 |

Kopā patērētas aptuveni patērētas 125 stundas un 45 minūtes. Šis rādītājs norāda, ka COCOMO aprēķins šī projekta ievaros neatbilst realitātei, ko varētu skaidrot ar to, ka projekts ir salīdzinoši apmēros neliels un to veica viens izstrādātājs. Tāpat jāņem vērā, ka šis laiks neietver laiku, kas tika izmantots koda optimizācijai,

8. Nobeigums

Kvalifikācijas darbā izstrādātā sistēma apmierina prasību specifikācijā izvirzītās prasības un atbilst specifikācijai. Sistēmas apjoms ir samērā neliels, taču tās modulārās struktūras dēļ to būtu salīdzinoši vienkārši attīstīt un papildināt ar papildus funkcionalitāti.

Izveidotais risinājums ļauj kontrolēt kameru neatkarīgi no lietotāja izvēlētās personālā datora operētājsistēmas, kam iepriekšēji risinājumi nav pieejami. Darba autors pēc papildus funkcionalitātes izstrādes plāno to padarīt pieejamu plašākas publikas lietošanai.

Projekta laikā izstrādātājs ieguva praktiskas zināšanas par informācijas sistēmu plānošanu un izstrādi, kā arī ieguva jaunas zināšanas par informācijas sistēmu izstrādes tehnoloģijām, piemēram, par *Chrome Native Messaging* tehnoloģijas iespējām un ierobežojumiem. Šīs iegūtās zināšanas būs noderīgas turpmākajā darba procesā un mācībās.

9. Definīcijas, akronīmi un saīsinājumi

API – Application Program Interface - lietojumprogrammas interfeiss.

CSS – Cascading Style Sheet – stila lapas kaskadēšana.

HTML – Hypertext Markup Language – hiperteksta marķēšanas valoda.

JSON – JavaScript Object Notation

Sistēma - Kopums, kopa, starp kuras elementiem, sastāvdaļām ir noteiktas attieksmes un kurai ir noteiktas funkcijas.

Pirmkods

ChromeMessageParser klase

```
class ChromeMessageParser():
```

```
    def __init__(self):  
        pass
```

```
    """
```

```
    Function used to convert input string value to a dictionary for further processing.  
    Returns a dictionary object.
```

```
    @:type message: string
```

```
    @:param message: Input message that was passed from chromeCommunicator
```

```
    @:rtype dict
```

```
    @:return Input parameter converted to a Python dictionary.
```

```
    """
```

```
    @staticmethod
```

```
    def convert_message_to_json_object(message):  
        output_object = json.loads(message)  
        return output_object
```

```
    """
```

```
    Function parses input message and returns a dict of either a parsed parameters for camera  
    request
```

```
    or returns a dict containing an error message.
```

```
    @:type in_message: dict
```

```
    @:param in_message: Data from chrome application
```

```
    @:rtype dict
```

```
    @:return Command (with or without command parameters if such are supplied) data for use  
    in camera communications
```

```
    """
```

```
    @staticmethod
```

```
    def process_message(in_message=None):
```

```
        if in_message and isinstance(in_message, dict):
```

```
            if "command" == in_message["type"] and len(in_message["properties"]) is not 0:
```

```
                command = in_message["properties"]["command"]
```

```
                if in_message["properties"]["command_params"] is not None and \
```

```
                    in_message["properties"]["command_params"] != "none":
```

```
                        command_param = in_message["properties"]["command_params"]
```

```
                        return_message = {"type": "command", "command": command, "parameters":
```

```
command_param}
```

```
            else:
```

```
                return_message = {"type": "command", "command": command, "parameters":
```

```
None}
```

```
        return return_message
```

```
    else:
```

```
        # @todo should be a warning instead
```

```
    return {"type": "error",
           "properties": {
             "error_code": 1,
             "description": "No command recieved"
           }}
  }
```

"""

Function that combines all other chrome message parsing functions.

@:type message = string

@:param message = Input message from chrome

@:rtype processed dict

@:return Message converted to necessary format

"""

```
def process_message_all(self, message=None):
    converted = self.convert_message_to_json_object(message)
    processed = self.process_message(converted)
    return processed
```

sendNativeMessage funkcija

```
/**
```

```
* Sends a message to server and updates the view.
```

```
* @param none
```

```
* @return none
```

```
*/
```

```
function sendNativeMessage() {
  var message = {"text": document.getElementById("input-text").value};
  port.postMessage(message);
  appendMessage("Sent message: <b>" + JSON.stringify(message) + "</b>");
}
```

Izmantotā literatūra

1. Pārlūkprogramma *Google Chrome* Pieejams: <http://www.google.com/chrome/>
2. Kameronas *Sony QX1* specifikācija. Pieejams: <http://store.sony.com/interchangeable-lens-style-camera-zid27-ILCEQX1/B/cat-27-catid-All-Cyber-shot-Q-Series-Cameras>
3. Extension package contents. Pieejams: <https://developer.chrome.com/extensions/overview>
4. Chrome NativeMessaging protokola apraksts. Pieejams: <https://developer.chrome.com/extensions/nativeMessaging>
6. Git versiju kontroles sistēma. Pieejams: <https://git-scm.com/>
7. Datu krātuve Github. Pieejams: <http://www.github.com>
8. Python PEP8 standarts. Pieejams: <https://www.python.org/dev/peps/pep-0008/>
9. Latvijas Valsts Standarts LVS 68:1996 „Programmatūras prasību specifikācijas ceļvedis”.
Latvijas Valsts Standarts LVS 72:1996 „Ieteicamā prakse programmatūras projektējuma aprakstīšanai”.
10. Sony lapa kameronas programmatūras izstrādātājiem. Pieejams: <https://developer.sony.com/develop/cameras/>
11. JSON Remote Process Call protokola dokumentācija. Pieejams: <http://json-rpc.org/>

Kvalifikācijas darbs „*Kameras kontroles programmatūra un Chrome spraudnis sistēmas stāvokļa pārraudzībai un kontrolei*” izstrādāts Latvijas Universitātes Datorikas fakultātē.

Ar savu parakstu apliecinu, ka darbs izstrādāts patstāvīgi, izmantoti tikai tajā norādītie informācijas avoti un iesniegtā darba elektroniskā kopija atbilst izdrukai.

Autors: *Kārlis Punka* _____ .05.2015.

Rekomendēju darbu aizstāvēšanai

Darba vadītājs/a: *B.dat, Nils Lindentāls* _____ .05.2015.

Recenzents: *M.dat. Egons Šolmanis*

Darbs iesniegts 01.06.2015.

Kvalifikācijas darbu pārbaudījumu komisijas sekretārs: *Darja Solodovņikova* _____

Darbs aizstāvēts kvalifikācijas darbu pārbaudījuma komisijas sēdē

____.06.2015. prot. Nr. _____

Komisijas sekretārs(-e): _____